

# 第4章 オペレーティングシステムの 構成

花田 英輔

(このPowerPointは渡辺名誉教授作成のものを花田が一部改編した)

1

## OSのためのハードウェア機能 (教科書4.1)

- ▶ OSがうまく動くために必要なハードウェア
- ▶ **実行モード**
  - OSだけが特権を持って動けるようにモードを切替える仕組み
- ▶ **割り込み**
  - OSが他プログラムの実行中でも動けるように割り込む仕組み

機能制限ON/OFF

途中に**割り込み**

2

## 実行モード

- ▶ OSも一つのプログラムである
  - それがお他プログラムの実行や資源割り当てを制御する

↓  
だから

- ▶ OSだけに制御を許すようなハードウェア機構が必要  
→ 「**特権**」モード

3

## 実行モードの切替

- ▶ 実行モード切替＝許容する命令範囲を場合により制限
  - **特権モード(カーネルモード)**
    - ・全ての命令が実行可能なモード
    - ・OSカーネルが実行するとき
  - **非特権モード(ユーザモード)**
    - ・特権命令は使えないモード
    - ・応用プログラムが実行するとき

4

## 特権命令とは

- ▶ うかつに実行すると、システム全体の正常動作を阻害する命令

例)

- 入出力機器の制御命令
- プロセッサのステータスを変更するような命令
- プログラムの実行制御命令

5

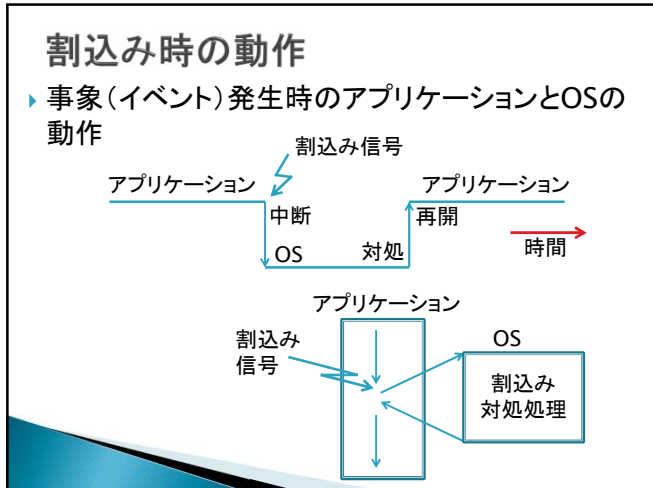
## 割り込み

- ▶ 他プログラムが動作中でも色々な**事象(イベント)**は発生する
  - 解釈不能な命令、アクセス不可なメモリ位置、キーが押される、マウスが動く、定期処理の時間が来る、許容されない特権命令、...

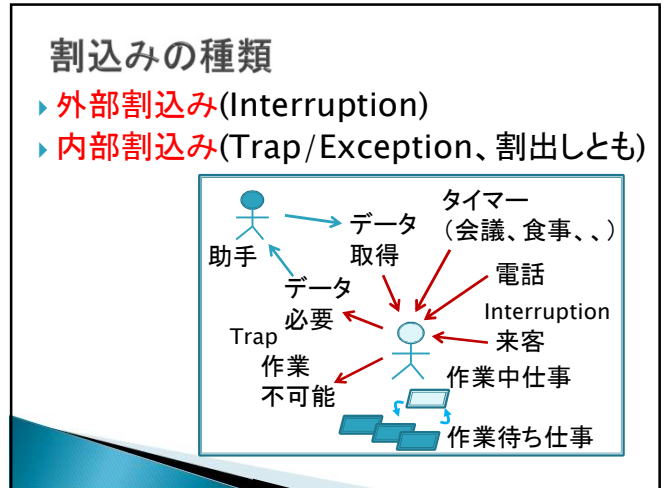
↓

- ▶ OSがこれらの事象を**検知して直ぐに対処**できるための**ハードウェア機構** = **割り込み**

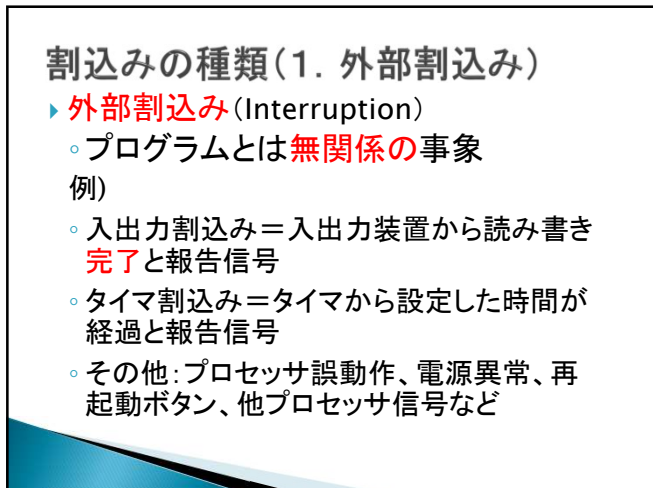
6



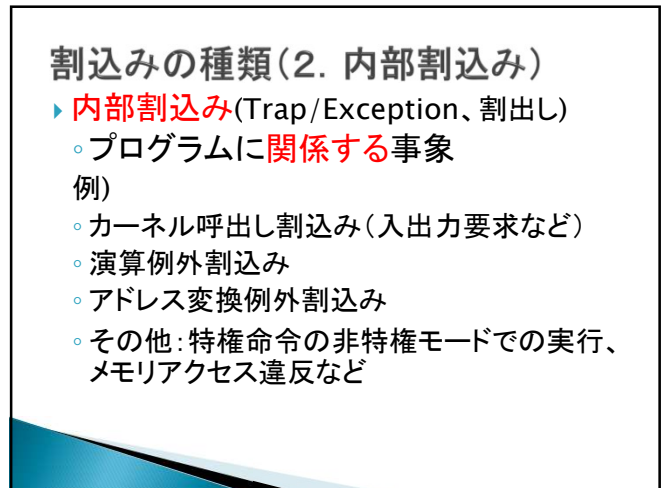
7



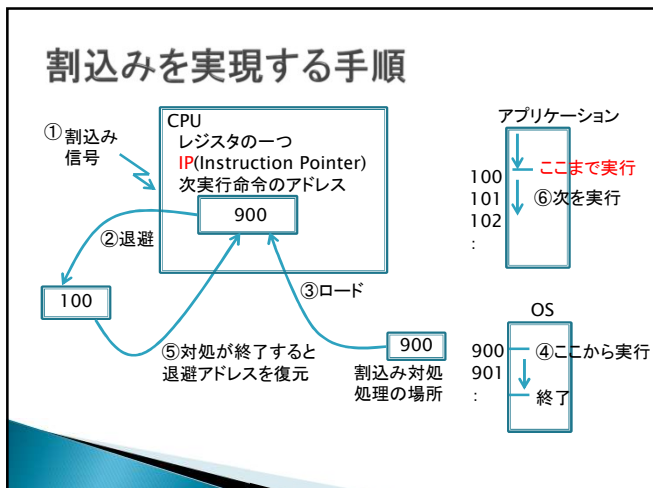
8



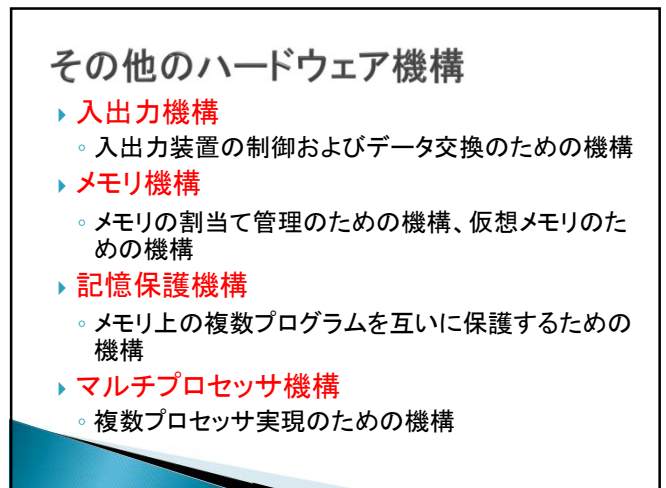
9



10



11



12

### 割り込みとマルチプログラミング(教科書4.2)

▶ **シングルプログラミング**

- 一つのプログラムを実行、入出力動作中はプロセッサが暇

↓

▶ **マルチプログラミング**

- 複数のプログラムを用意しておいて、プロセッサが空いたら切り替えて実行すれば効率的

13

### マルチプログラミングにおける切り替え

入出力要求割り込み (プログラムAが発生)

入出力終了割り込み (入出力装置が発生)

プログラムA実行

プログラムB実行

OS

OS

時間

装置へ入出力開始命令

開始

入出力装置動作

終了

割り込み

割り込み

割り込み

A

B

C

A

OS

OS

OS

14

### OSの核:カーネル(教科書4.3)

▶ **カーネル** = 特権モードで動くOSの主要部分

- 危険な処理はここに置く(ハード制御など)
- そうでない処理は外でも可

応用プログラム 非特権モード

OS カーネル以外 非特権モード

カーネル 特権モード

ハードウェア

具体的なカーネルの構成例は教科書 p.51を参照のこと

15

### 実行モードの切替え

- ▶ 特権モードから非特権モードへの切替え命令 = **有り**
- ▶ 非特権モードから特権モードへの切替え命令 = **無し**  
⇒ 存在すると誰でも特権を得ることが可能
- ▶ **ではどうする?**  
⇒ 割り込み発生時に、同時に特権モードへの切替えを実行
- 割り込み処理部分は特権モード、処理後に非特権へ

特権モード

非特権モード

マシン命令有り

マシン命令無し

割り込み

16

### 割り込み処理の流れ

割り込み  
カーネル呼出し割り込み、演算例外割り込み、入出力完了割り込み、タイマ割り込み……

割り込み

カーネル呼出し割り込み、演算例外割り込み、入出力完了割り込み、タイマ割り込み……

呼出しコードの判定

カーネル

終了処理

エラー処理

入出力要求処理

入出力完了処理

演算例外処理

……

カーネル出口処理 (元のアプリケーションへ or 別のアプリケーションへ)

17

### カーネル処理中に割り込みが届くと?

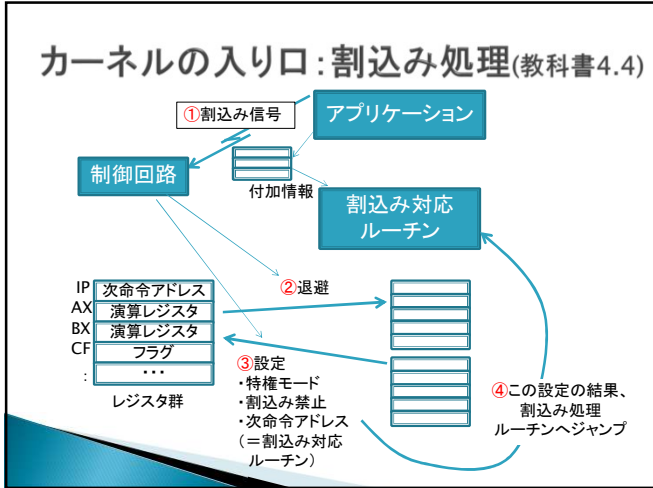
- ▶ 割り込み処理を中断して別の割り込み処理??
- ▶ **単純な解決**
  - カーネル処理中は割り込み禁止
  - 「今忙しいからダメ、帰って」
- ▶ より**高度な解決**
  - 割り込みを受け付け処理だけして、前の処理が終了後に処理
  - 「後でやるから、受付簿に記入して待ってて」

18

### カーネルに含まれる処理

- ▶ プロセスの実行管理
- ▶ 入出力装置への入出力の管理
- ▶ ファイルの管理
- ▶ メモリ領域の管理
- ▶ 仮想メモリの制御
- ▶ プロセスの同期制御
- ▶ タイマの管理
- ▶ 割り込み処理

19



20

### OSのカーネル以外の部分(1)(教科書4.5)

- ユーザインターフェースプログラム**
  - CUI: 利用者からコマンドを受け取り、解釈して対応プログラムを起動
    - = コマンドインタプリタ、UNIXではシェル(貝殻)
  - GUI: デスクトップ描画・マウス操作等のためのプログラムあり
- サービスプログラム**
  - OSの一部として提供
  - 応用プログラムの位置づけ
    - ・ コンパイラ・インタプリタ環境など

21

### OSのカーネル以外の部分(2)(教科書4.5)

- システムプロセス用プログラム**
  - システム内で自動起動し、システムのプロセスとして動くもの
  - UNIXではデーモン、Windowsではサービスと呼ぶ
    - ・ ネットワークサーバ機能等
- API用ライブラリ**
  - システムコール関数などをライブラリとして提供
    - ・ 利用者プログラムに結合
  - ▶ OSは一つのプログラムではなく、多くのプログラムの集合体
    - 範囲は時代により変化
      - ・ コンパイラはOS内から外へブラウザはOS外から内へ

22

### マイクロカーネルとモノリシックカーネル(教科書p.51のコラムを拡張)

- ▶ **モノリシックカーネル**=OSの構成要素を単一のメモリ空間で実行、旧来のOS
  - ↓ より新しい考え方
- ▶ **マイクロカーネル**=OSを構成する幾つかの要素・機能をカーネル空間から切り離し外部モジュール化
  - ⇒ 危険な部分を最小化できる
  - ↓ しかし処理切替の負荷大
- ▶ **ハイブリッドカーネル**=折衷型
  - Linux モノリシックカーネル→モジュール化して分離が進む
  - Windows マイクロカーネル→効率必要部分を取込み増大

「モノリシック(monolithic)」は「一枚板」という意味

23

### 今回の課題

1. 割り込みとは何か。またその発生要因にはどのようなものがあるか説明せよ。また、発生要因別に「内部割り込み」か「外部割り込み」かを記せ。
2. (予習)入出力を効率化する手法について調べて記せ
  - ▶ 今回のファイル名は「学籍番号-OS05.docx」(例:22238000-OS05.docx)としてください
  - ▶ 締切:11月10日(金) 18:00 (遅れた場合は減点)

**記載時の注意事項**

- ▶ レポート内にも名前を書くこと
- ▶ 参考資料(Webページ)がある場合は**出典を書くこと**
- **出典を書かずに引用した場合は減点対象です**

24